Industrial Protocols for Spacecraft Command and Control

by
Wm. Randy Heuser
Member of the Technical Staff
Jet Propulsion Laboratory
California Institute of Technology
Mail Stop 301-235
4800 Oak Grove Drive
Pasadena, California 91109
Office phone (818) 354-0956, Fax (818) 354-9068
email: rheuser@binky.jpl.nasa.gov

Abstract

A standard data interface for spacecraft and intelligent spacecraft devices will have a major impact on the cost of building and flying satellites. The Space Messaging Service is a key element in a proposed "open" standard for Space Project Mission Operations Control Architecture (SuperMOCA). The rationale for a messaging service is examined. The selection of the Manufacturing Message Specification and Fieldbus standards as the initial basis for a comprehensive set of Space Messaging Services is discussed. Some modifications to the industrial standards required to address the space environment are proposed.

Introduction

In the initial phase of Space Station Freedom, a major effort was launched to establish a comprehensive information management architecture for operation of the spacecraft. A number of innovative concepts were developed during that period of time. Early design efforts determined that a messaging service on top of a packet transport service, would greatly enhance the operation of the Space Station. However, the 1990s saw Space Station Freedom scaled back from its original scope and many of these innovative concepts were dropped. Despite the re-direction of Space Station, a small group of interested parties continued to meet and discuss the challenges of a comprehensive command and control system for space missions. The Space Project Mission Operations Control Architecture (SuperMOCA) has emerged from those discussions as a effort to develop an "open specification" for civil and military space projects. This article will discuss one key element of the architecture, the Space Messaging Service.

Figure 1 provides a graphical representation of the five basic elements in the Mission Operations Control Architecture. The five elements are:

- The Data Architecture provides for the organization and management of the abstract objects that represent the real assets and resources of the spacecraft.
- The System Management Architecture provides for the organization and management of the operations environment of the spacecraft.
- The Control Interface provides a user friendly mechanism to operate and manage the spacecraft.
- The Decision Support Engine operates the spacecraft and processes commands for the safe execution of the mission. The Decision Support Engine operates on the abstract objects and environment constraints contained in the Data Architecture and System Management Architecture databases.
- The Space Messaging Service provides a comprehensive set of inter-processor communication services for ground-to-space[1] and space-to-space messaging.

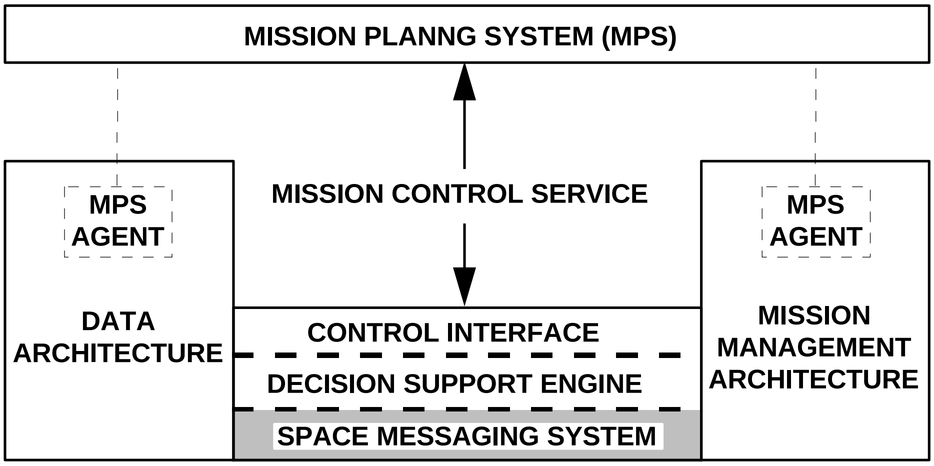


Figure 1. The five basic elements of the Space Project Mission Operations Control Architecture.

Manufacturing Message Specification (MMS) was the subject of intense investigation in the early days of Space Station Freedom. Developed as an “open” industrial standard for control of robots and intelligent factory devices, analysis for Space Station Freedom determined that MMS encompassed all of the services required for the operation of the space station. Rather than create a completely new set of

[1] Ground-to-space in this paper refers to the ground-to-space uplink and the space-to-ground downlink.

communication services, SuperMOCA will build on the industrially proven capabilities of MMS to develop services ideal for the spacecraft environment. The Space Messaging service will also draw on the emerging Fieldbus specification which is derived from the MMS standard.

Problem Overview

A spacecraft is basically a robot operating in the hostile environment of space. Some spacecraft are simple machines with no moving parts and passive sensors gathering data. Other spacecraft are complex electro-mechanical devices operating in deep space, far from earth. And still other spacecraft are complex systems populated by astronauts. Regardless of simplicity or complexity, the robot model is a good fit to the spacecraft paradigm. The Space Messaging Service (SMS) envisioned for SuperMOCA would provide a set of services to meet the requirements of this diverse array of spacecraft. In addition to providing services for ground-to-space communications, the Space Messaging Service will also provide services to support spacecraft-to-spacecraft communications and onboard-system-to-onboard-system communications (Figure 2).

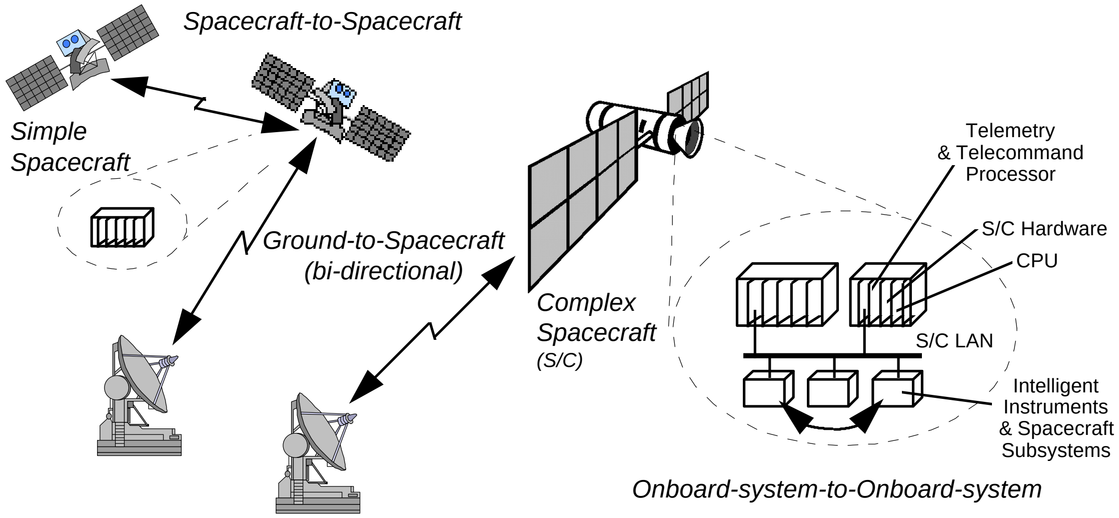

Spacecraft-to-Spacecraft
Simple Spacecraft
Ground-to-Spacecraft (bi-directional)
Complex Spacecraft (S/C)
Telemetry & Telecommand Processor
S/C Hardware
CPU
S/C LAN
Intelligent Instruments & Spacecraft Subsystems
Onboard-system-to-Onboard-system

Figure 2. Communication paths supported by the Space Messaging Service.

The Space Messaging Service is sandwiched between the Decision Support Engine and the lower layers of the communications stack (see Figure 3). The messaging service defines a set of messages with structure and syntax to support a dialog between systems. The message type, structure and syntax makes possible the construction of software that can generate and interpret messages. And that software can be reused from mission to mission. Furthermore, the message structure and syntax makes it possible to examine and evaluate messages to determine their impact on the spacecraft.

Why do we need a messaging service? Why not simply provide a packet exchange service and leave message definition up to the specific spacecraft implementation? At first, this may seem a reasonable approach because packet based systems are a common practice today. However, the creation of new message schemes for each mission is a very expensive proposition. In addition, it makes each spacecraft unique and forces the development of a unique mission operations facility.

The need for a messaging service can be illustrated through simple business scenarios that many people have experienced. Consider a small company that needs to order parts from another small company around the corner. The owner of the Acme Valve Company writes a note ordering five small widgets. The note is placed in an envelope and hand delivered to the Best Widget Company. The owner of Best Widget reads the note, pulls the requested widgets out stock and delivers the widgets to Acme. In this scenario, the packet is the envelope that contains the hand written note. It is the note that contains the information requesting five widgets. Since individuals generate and interpret the information carried in the packet, there is no need for a formal structure that insures the correct interpretation of the information. And since the note and the widgets are delivered directly, there is no need for addresses (see Figure 4).

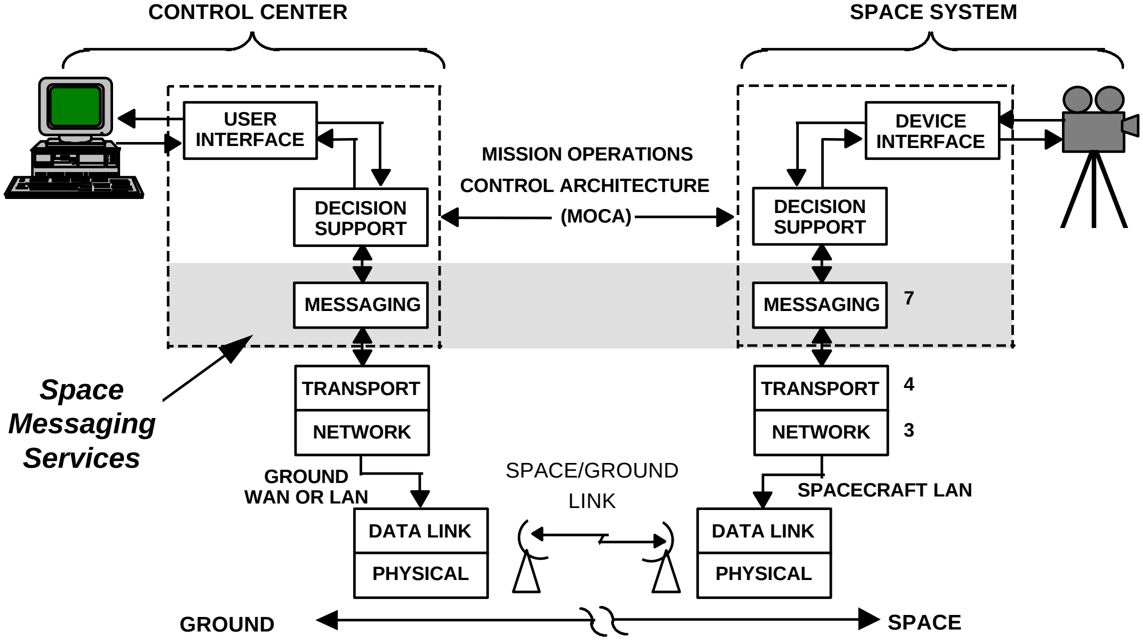
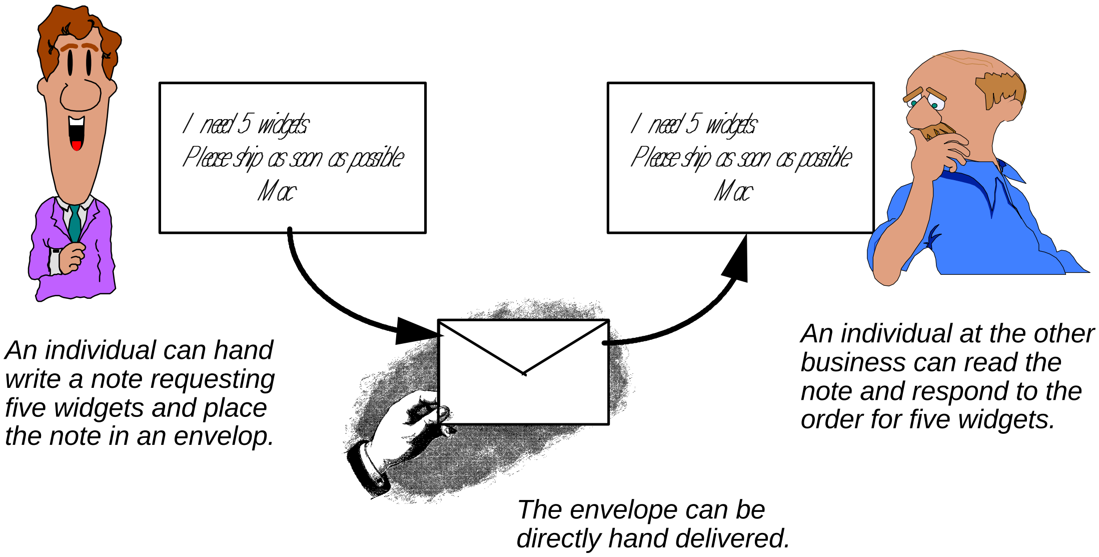


Figure 3. The space process control system operates over a communications stack.



I need 5 widgets
Please ship as soon as possible
Mac

I need 5 widgets
Please ship as soon as possible
Mac

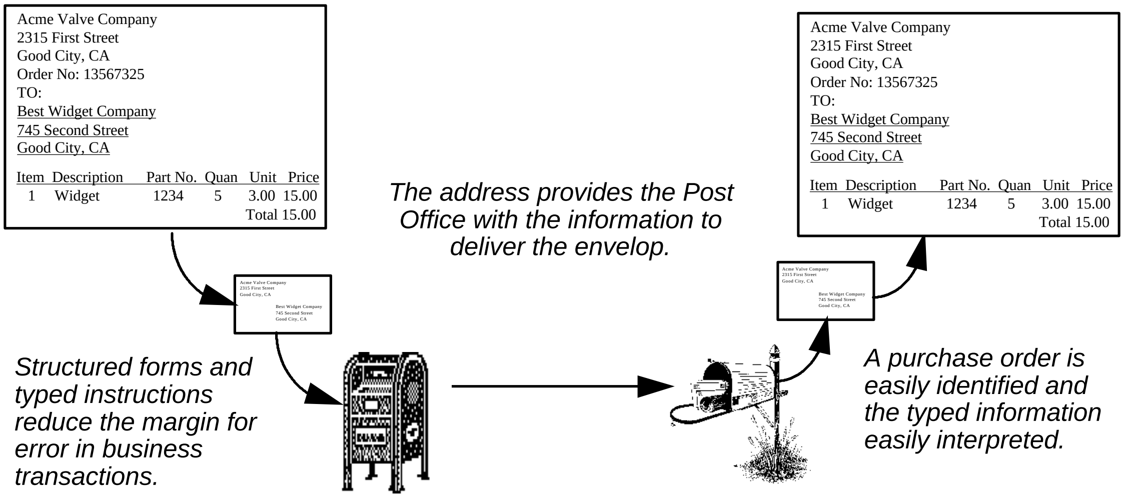
An individual can hand write a note requesting five widgets and place the note in an envelop.

An individual at the other business can read the note and respond to the order for five widgets.

The envelope can be directly hand delivered.

Figure 4. In small systems, individuals can exchange and interpret information easily.

Business has been good and now the Acme Valve and Best Widget Companies have moved to larger facilities and have many employees. Purchase order forms are used and orders typed with specific part numbers. The order is placed in an envelope, addressed and mailed. Again, the packet is the envelope that contains the purchase order. The destination and return addresses are used by the Post Office to deliver the envelope to the correct destination. The contents of the envelope is of no concern to the Post Office. The envelope could just as well contain a bill or a letter. The purchase order form identifies the general contains of the envelope and the detailed structure identifies the specific items ordered. The purchase order form assists in the interpretation of the items and quantities ordered (see Figure 5).



Acme Valve Company
2315 First Street
Good City, CA
Order No: 13567325
TO:
Best Widget Company
745 Second Street
Good City, CA

Item	Description	Part No.	Quan	Unit	Price
1	Widget	1234	5	3.00	15.00
				Total	15.00

Acme Valve Company
2315 First Street
Good City, CA
Order No: 13567325
TO:
Best Widget Company
745 Second Street
Good City, CA

Item	Description	Part No.	Quan	Unit	Price
1	Widget	1234	5	3.00	15.00
				Total	15.00

The address provides the Post Office with the information to deliver the envelop.

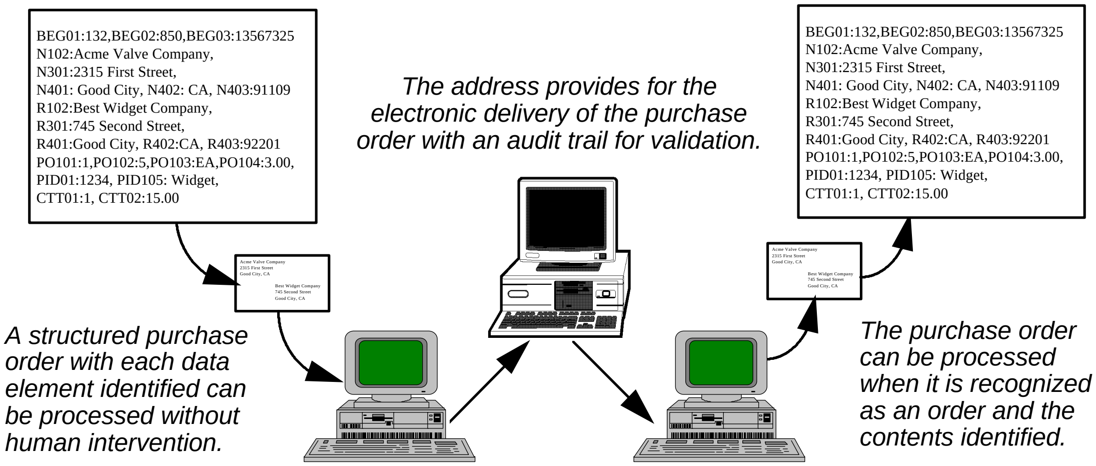
Structured forms and typed instructions reduce the margin for error in business transactions.

A purchase order is easily identified and the typed information easily interpreted.

Figure 5. Systems become more formal as businesses grow and increase in size.

Now, business has been so good that the companies are automating their factories. To automate the ordering process, each business obtains a computer with software designed to create electronic versions of the purchase order. All of the information contained in the order is formulated into a message that can be interpreted by a machine. The company placing the order and the company receiving the order must be clearly identified. The items to be ordered, their descriptions, part numbers, quantities and prices must also be identified. The order is transferred electronically to the vender as a series of data packets through a Value Added Network (VAN).

It is easy to understand how and why businesses move from hand written notes to typed business forms. Small businesses operate on a personal relationship level. But as businesses grow, it becomes difficult for everyone in a business to have a personal relationship with all the customers. In the example above, business forms replaces the hand written note to assist individuals in interpreting the needs of the customer. The envelope is used to transport the purchase order through the Post Office. It does nothing to identify the type of message or information contained in the envelope. Similarly, the electronic form identifies the message as a purchase order and identifies each element of the contents. The electronic packets are addressed and used to transfer the message to the correct destination. In short, packet based systems support the transport of data, while messaging services provide structure and meaning to the data contained in the packets (see Figure 6).



```
BEG01:132,BEG02:850,BEG03:13567325
N102:Acme Valve Company,
N301:2315 First Street,
N401: Good City, N402: CA, N403:91109
R102:Best Widget Company,
R301:745 Second Street,
R401:Good City, R402:CA, R403:92201
PO101:1,PO102:5,PO103:EA,PO104:3.00,
PID01:1234, PID105: Widget,
CTT01:1, CTT02:15.00
```

```
BEG01:132,BEG02:850,BEG03:13567325
N102:Acme Valve Company,
N301:2315 First Street,
N401: Good City, N402: CA, N403:91109
R102:Best Widget Company,
R301:745 Second Street,
R401:Good City, R402:CA, R403:92201
PO101:1,PO102:5,PO103:EA,PO104:3.00,
PID01:1234, PID105: Widget,
CTT01:1, CTT02:15.00
```

The address provides for the electronic delivery of the purchase order with an audit trail for validation.

A structured purchase order with each data element identified can be processed without human intervention.

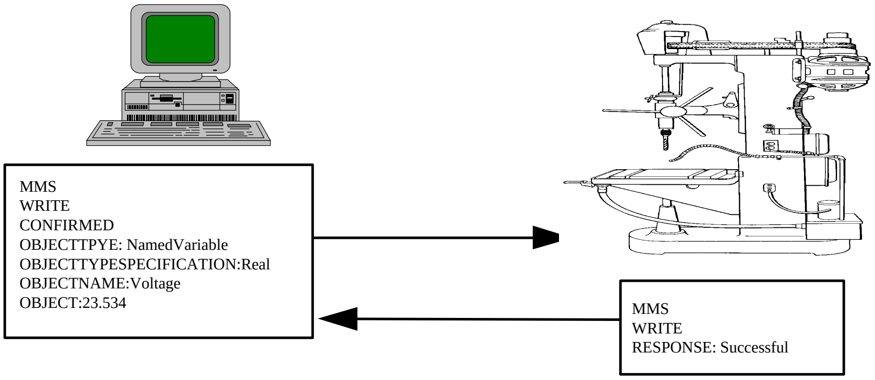
The purchase order can be processed when it is recognized as an order and the contents identified.

Figure 6. Automation requires structure and definition to ensure the correct interpretation of information. (Note: the example is intended as instructional and does not adhere strictly to the X12.1 - 850 specification)

A set of standards has evolved to support the exchange of the business messages described above. These are collectively known as the Electronic Data Interchange (EDI) standards. While significantly different from the messaging services required to operate a spacecraft, the EDI approach demonstrates several key characteristics required in many messaging services. First, there are specific message types used to conduct business and these are frequently organized into message groups. For example, the Purchase Order (850) is a specific message type that is a member of a group that includes: the Purchase Order Acknowledgment message (855), the Routing and Carrier Instructions message (853), the Ship Notice / Manifest message (856), the Shipment and Billing Notice message (857) and several others.

Second, each message is self-identifying. Every data item in the message has a label to identify the type of data that follows the label. Each message type has an associated data dictionary that specifies the labels used in that message. A third characteristic is that messages reflect the relationship between the parties involved. In the example above, the consumer (or client) issues the Purchase Order and the supplier (or server) responds with the Purchase Order Acknowledgment.

The same characteristics found in EDI are also found in protocols like MMS and Fieldbus (SP-50) which were designed specifically for industrial process control. First, there are specific types of messages organized into groups that operate together to provide services. The MMS Write, Read and Information Report services are all members of the Variable Management service set. Second, MMS and SP-50 messages are self-identifying. Like the EDI Purchase Order, the MMS Write service identifies the data in the message. And third, MMS and SP-50 are modeled on a relationship between the two parties exchanging messages. The “client-server” model is the basis for MMS and the “publish-and-subscribe” model is the foundation for the Fieldbus. The MMS client-server relationship and self-identifying messages are depicted in Figure 7.



Each MMS message is structured with all elements identified to support interruptation by the receiving system.

Many MMS messages require a confirmation from the receiving system.

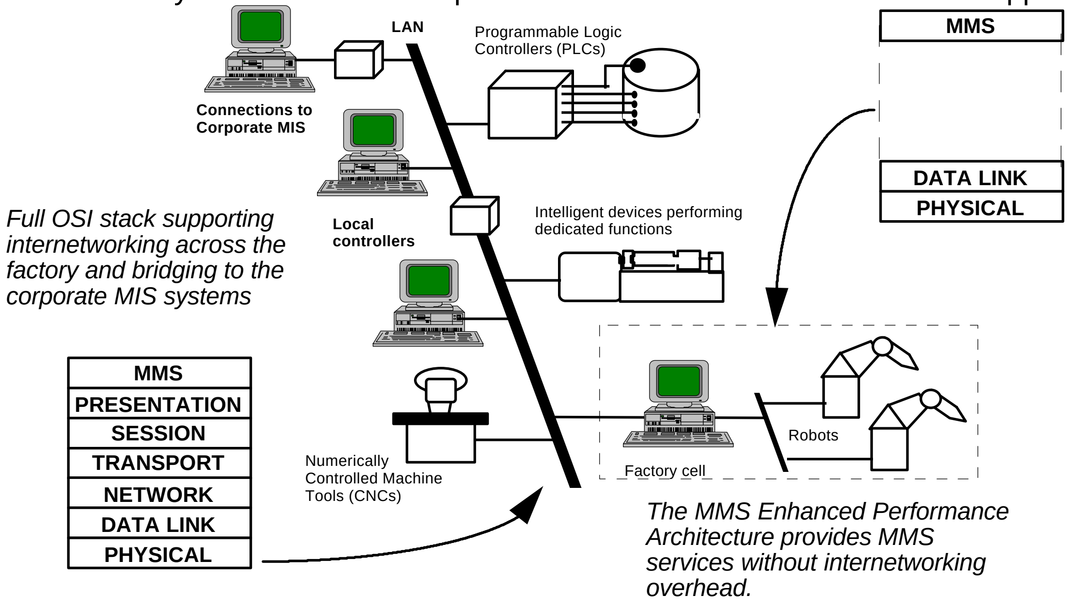
Figure 7. Like EDI, MMS messages are self-identifying. (Note: the example is intended as instructional and does not adhere strictly to the ISO 9506 specification)

Industrial Solutions

MMS was designed to support the automation of factories. Many factories today employ intelligent devices to perform complicated operations without human intervention. Some of these devices are: robots, numerically controlled tool systems (lathes and mills) and programmable logic controllers (PLCs). Figure 8 depicts the environment where the services of MMS were designed to operate. As an “open” specification, MMS provides the manufacturing sector with an automation solution where products from different vendors can be integrated into a factory with little or no traditional software development. MMS can be implemented over a full protocol stack to provide a fully integrated factory system. Using the full stack, network segments can be linked together with commercial network routers, and the factory floor linked to the corporate Management Information System. The MMS Enhanced Performance Architecture (EPA) was developed to support factory cells which require high performance data exchange and do not require direct access to other factory devices or the corporate information system.

There are 87 services provided by MMS to manage intelligent devices across a network[2]. A vendor's product will identify itself with the product name, serial number and version number in response to the MMS Identify service request. The device will also provide its list of managed objects, its list of capabilities and its list of supported parameters in response to the corresponding MMS service request. And programmable devices can have programs downloaded across the network, initiated and managed using the MMS domain and program invocation services.

The Fieldbus specification is being designed for devices operating in critical process control systems like chemical plants and refineries. The Fieldbus will support tight, critical control loops and operate directly over a data-link layer designed for environments with explosive potential. As such, the Fieldbus is not intended to operate over a wide area network. A local controller will manage the Fieldbus and operate as a bridge to corporate Management Information Systems (see Figure 9). The sensors and actuators employed in chemical processing plants are candidates for Fieldbus implementations. These devices are relatively simple when compared to the type of manufacturing devices that would use MMS. As such, the developers of the Fieldbus have incorporated only the basic MMS services into the Fieldbus specification. Like MMS, Fieldbus provides service for the device to identify itself across the network. And


LAN
Programmable Logic Controllers (PLCs)
Connections to Corporate MIS
Local controllers
Intelligent devices performing dedicated functions
Numerically Controlled Machine Tools (CNCs)
Robots
Factory cell
MMS
PRESENTATION
SESSION
TRANSPORT
NETWORK
DATA LINK
PHYSICAL
MMS
DATA LINK
PHYSICAL

Full OSI stack supporting internetworking across the factory and bridging to the corporate MIS systems

The MMS Enhanced Performance Architecture provides MMS services without internetworking overhead.

Figure 8. MMS was developed to support the automation of factories systems.

[2] A summary of MMS services is provided in Appendix A.

Fieldbus provides services to monitor the status of devices and control the operation of actuators. However, the Fieldbus specification goes beyond MMS and provides for an embedded data dictionary to support integration of the Fieldbus devices into the network.

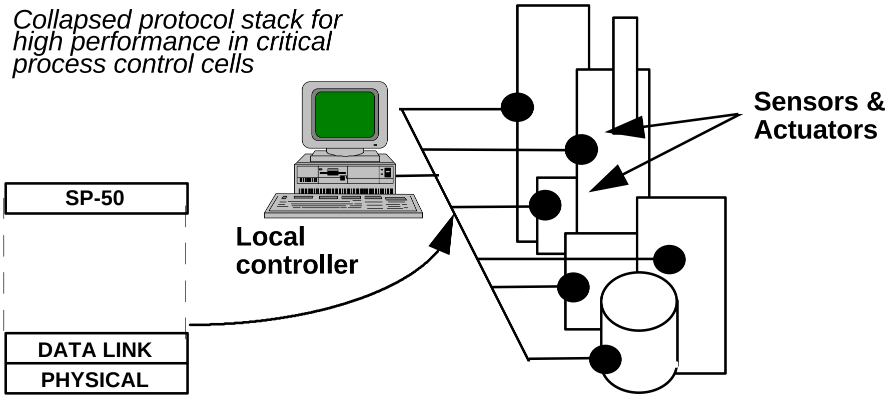

Collapsed protocol stack for high performance in critical process control cells
Sensors & Actuators
SP-50
Local controller
DATA LINK
PHYSICAL

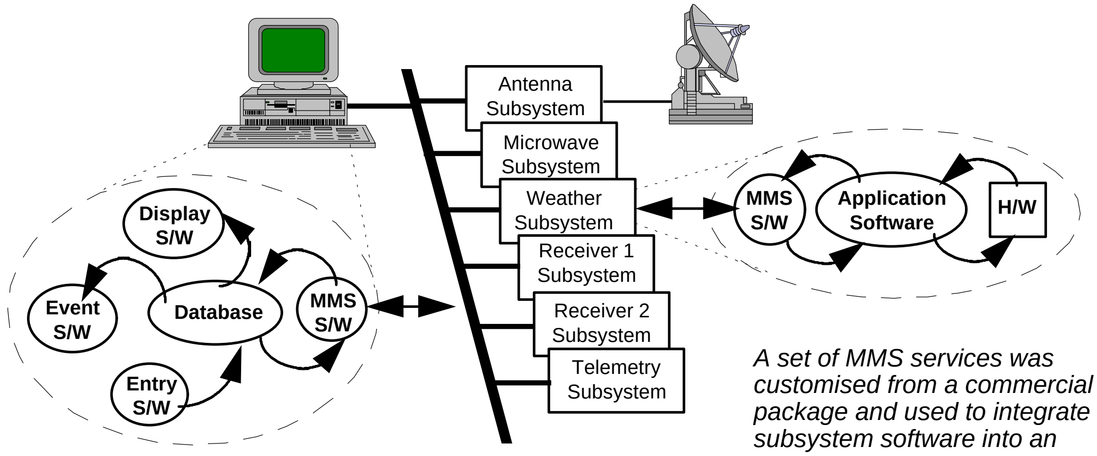
Figure 9. The Fieldbus is being developed to support automation in facilities with tight process control loops. The network is designed to be used in environments with explosive potential.

A Real World Example

In many ways, a tracking station is like a factory. Today's tracking stations employ intelligent devices to perform complicated operations with a minimum of human intervention. The tracking station antenna is a robot that moves under computer control to acquire the signal from a spacecraft and follow that spacecraft across the sky. The operation of receivers, transmitters and various signal processing equipment is very similar to the way Programmable Logic Controllers operate motors, thermocouples, relays and switches in a factory.

This view was put to the test in Deep Space Station 13 (DSS-13), the Deep Space Network's experimental tracking station at Goldstone, California. The monitor and control system was built using MMS services over a local area network. A workstation was installed as a Station Control System to manage the station resources. Each element (or subsystem) of the station is operated through a computer. The antenna, receivers, total power radiometer, microwave switching and weather station were all computer controlled. A commercial MMS package was tailored to slip into each subsystem to provide network communications (see Figure 10).

The subsystem controllers execute programs to operate their connected (or dedicated) hardware. The subsystems operate autonomously and function as MMS server with respect to the Station Control System. The Station Control System operates as an MMS client to configure, control and monitor the operation of the subsystems. For example, the Station Control System will connect to the antenna controller, configure the controller for a spacecraft track, initiate the antenna controller to track the spacecraft and monitor the antenna controller through the activity. The antenna controller responds to the Station Control System's commands and provides performance data as required (see Figure 11).



A set of MMS services was customised from a commercial package and used to integrate subsystem software into an MMS environment.

A commercial Supervisory Control and Data Acquisistion (SCADA) software package was integrated to support MMS control of the station subsystems. The SCADA package provided most of the functions required to operate the station.

Figure 10. Deep Space Station -13 employs MMS services to monitor and control all of the station subsystems from a single workstation.

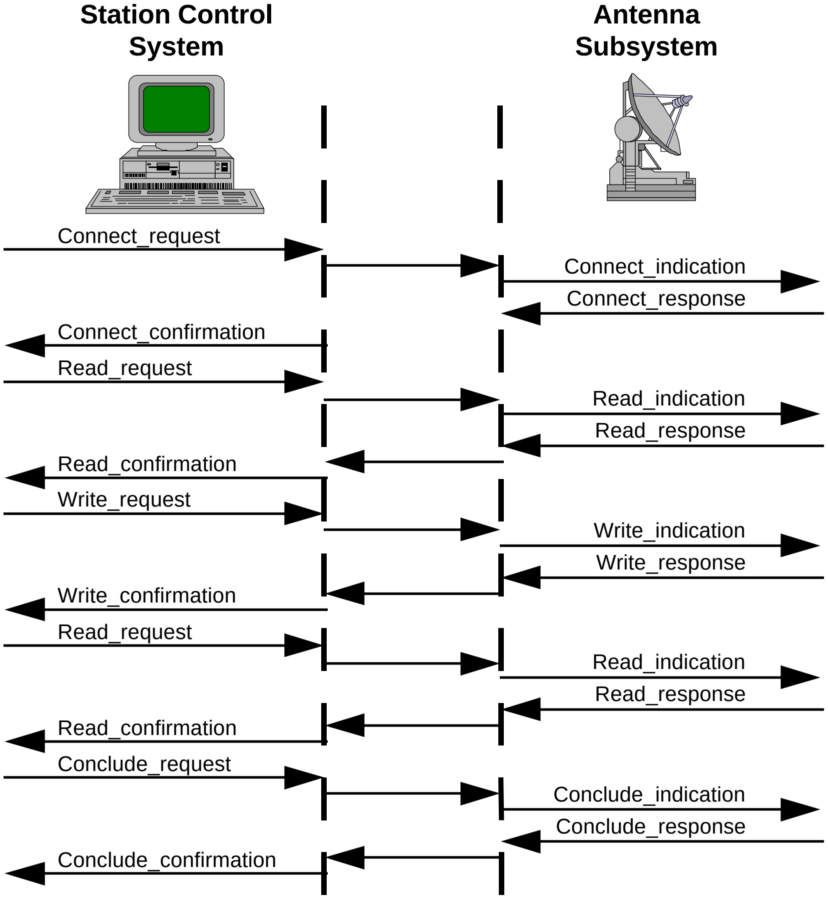


Figure 11. The Station Control System connects to the antenna, reads its current configuration and writes a new configuration to the antenna control subsystem to initiate its operation. The Station Control System monitors performance of the antenna by reading its status and disconnects from the antenna at the conclusion of the activity.

The subsystems can also operate as a client to other subsystems. For example, the antenna controller requires measurements of received power from the total power radiometer. The antenna controller can connect to the total power radiometer controller and read the received power as necessary. Once configured and initiated, the entire station will operate during a spacecraft pass without invention from the Station Control system and will continue to operate should the Station Control System fail and return to service at some later point in time.

A key part of the Station Control System was the incorporation of a commercial software package developed for factory automation. This product provides the man-machine-interface, device communications, event reporting, performance monitoring, report generation and a variety of associated automation services. The product implemented at DSS-13 represents a class of products frequently referred to as “Supervisory Control and Data Acquisition” (SCADA) or “Application Enablers” software packages. These products are designed to be installed and configured through table entries, eliminating most of the traditional software development effort. MMS based device communications is well designed for table driven software packages. Using the combination of MMS and SCADA products, substantial cost savings were achieved over “traditional” approaches. In addition, the application of commercial products provided enhanced reliability through its large customer base. When the DSS-13 was initiated, only two companies building SCADA products provided MMS device drivers for their products. Today, six companies offer MMS device drivers for their products.

The Ground-to-Space Communications

In many ways, control of a spacecraft is very similar to the control of a robot on the factory floor. A factory robot is composed of motors, actuators and sensors that operate in concert to produce real products. Similarly, spacecraft are collections of sensors and electro-mechanical devices that operate in space performing real tasks. The services of MMS provide support to establish communications with the factory robot, determine the robot’s identity, and status, and control its operation. The following scenarios are offered as examples of how an MMS based Space Messaging Service (SMS) might function.

Consider ground-to-space communications for a simple spacecraft in near earth orbit. A ground station would establish a connection with the spacecraft and request its “identity” and “status”. The “identity” and “status” data returned would be used by the ground system (both mission control and ground tracking station) to verify connection to the correct spacecraft and determine its current state. A series of SMS Reads by the ground system obtain detailed information about the configuration, health and status of the spacecraft. Simultaneously, recorded data would be transferred to the ground using the File Management services and new programs for instrument operations are uploaded using the Domain Management services. As the communication session nears it end, mission control would perform a series of SMS Writes to reconfigure the spacecraft based on real-time evaluation of health and status data (see Figure 12).

Onboard Communications

A spacecraft is like a self-contained, mini-factory. A spacecraft has its own power-plant and environmental (thermal) control subsystems. And the onboard instruments produce the spacecraft (mini-factory) product, the data. To understand the application of an MMS based Space Messaging Service (SMS) onboard a spacecraft, consider a spacecraft composed of intelligent subsystems and instruments interconnected across a computer bus or local area network.

The spacecraft instruments, the power subsystem, the thermal control subsystem and all of the other subsystems would be built with intelligent controllers designed to interface as servers on the bus or local area network. The subsystem controllers would reflect their status and performance through variables accessible by SMS Variable Read services. The subsystems would be configured and controlled through SMS Variable Write and Program Invocation services. And the subsystems would employ SMS Event services to reflect their changing state.

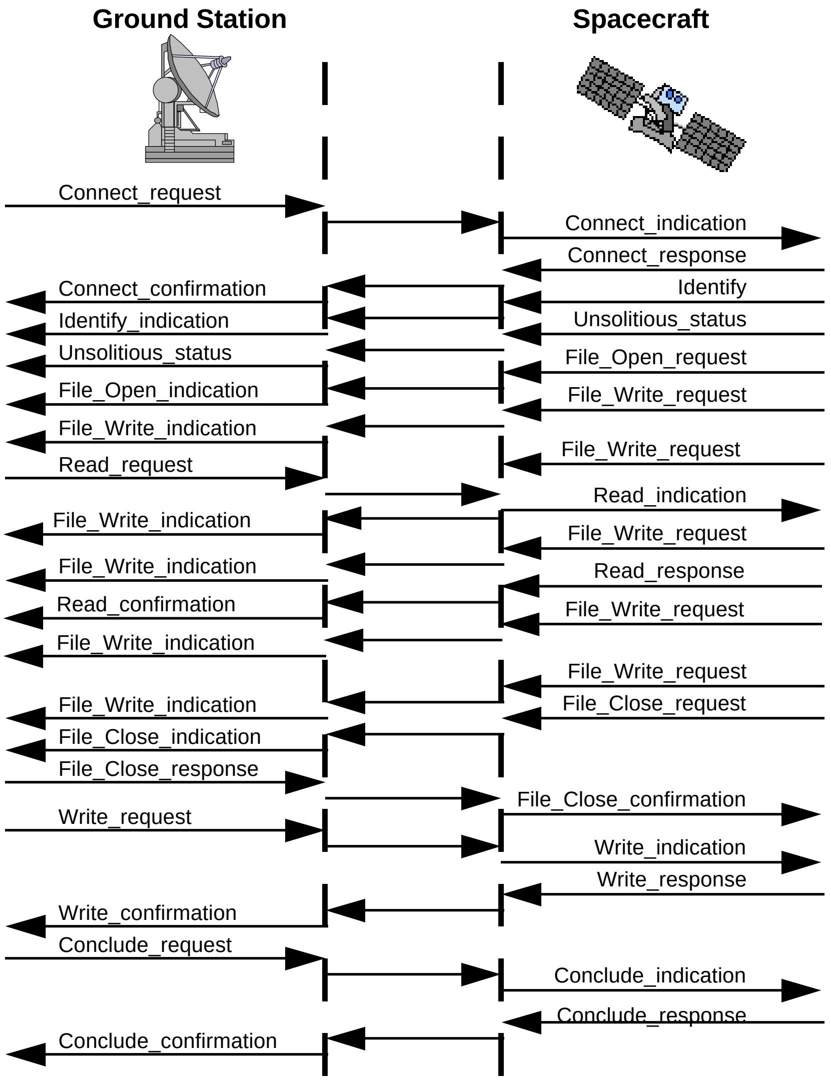


Figure 12. The ground system connects to the spacecraft. The spacecraft identifies itself, reports its status and begins to transfers its data product as a file. The ground station requests additional status information from the spacecraft and writes new configuration data before concluding the activity.

The spacecraft central control system would employ a SuperMOCA Decision Support Engine and operate as a client to the instruments and subsystems. The central control system would obtain the status and performance of subsystems and instruments using SMS Status and Variable services. The central control system would operate the subsystems and instruments using SMS Variable Write and Program Invocation services. The control system would enroll in the instrument defined SMS Events to receive notification of their changing state. And the control system itself would receive new instructions in flight through SMS Domain Upload services.

Putting it all Together

The goal today is to build spacecraft “faster, better and cheaper”. To achieve this goal, we must create an environment where the cost of developing and building spacecraft can be amortized over many missions. If the common elements required by most spacecraft could be built as scaleable, off-the-shelf components, the manufacturers of those devices could amortize the development costs over many spacecraft. Spacecraft built from off-the-shelf components will require standard physical, electrical and data system interfaces to support their integration into the vehicle. MMS has been used to build intelligent factory devices that can be integrated with little or no software development. Similarly, a Space Messaging Service would provide spacecraft developers with a similar data system interface for their intelligent devices.

An intelligent spacecraft component would be delivered with a SMS data system interface and an instruction book. The instructions would describe its operation and explain how to monitor and control the device through its SMS interface. The manufacturer would not deliver source code, program stubs or remote procedure calls. And the spacecraft developer would not need to integrate, compile or link software from many manufacturers into the spacecraft command system. Instead, SuperMOCA Decision Support Engine would be configured to operate the spacecraft components through Space Messaging Services.

The standards do not impose a rigid structure on manufacturers, but can expand the flexibility of their products. And the standards will not stifle the manufacturer’s creativity, but re-direct that creativity toward better products. In the future, two vendors of spacecraft power subsystems might compete for selection by offering SMS data system interfaces. One might provide a basic subsystem interface while the other advertises a sophisticated, event driven, self-diagnostic subsystem interface. For this example, let us assume that the project selects the sophisticated power subsystem for its spacecraft. However, the launch schedule prevents the project from implementing the sophisticated features and only the basic functions are integrated during spacecraft assembly.

After launch, anomalous thermal conditions are identified onboard the spacecraft that threaten the mission. The mission controllers decide to use the advertised SMS event and diagnostic services to determine if the power subsystem is operating as required. Mission control instructs the central control system to enroll in the power subsystem SMS Event services associated with “exceeding the maximum operating temperature”. The central control system is also instructed to invoke the power subsystem’s diagnostic program when the “maximum operating temperature” event notice is received. The results of the event and program invocation services assist mission controllers with fault isolation.

The discussion above illustrates the potential impact of a standard Space Messaging Service. With a standard data system interface, the manufacturers of spacecraft components can focus on building devices instead of unique interfaces. With standard interfaces, software components will be developed for use on mulitple spacecraft. And with standard interfaces, the ground support facilities will have a common basis to support all spacecraft. The services of MMS provide a foundation for a Space Messaging Service that is robust and flexible.

The Influence of Space Station

Space Station had a major influence on the selection of MMS as the basis for SMS. And a key issue for Space Station was controlled access to station resources. While the developers of the Space Station information system wanted to provide ground based users with as much access as possible, the safe operation of Space Station and the safety of the astronauts was of paramount concern. The anticipated data rates and real-time control requirements made evaluation of individual messages for their impact on Space Station operation impossible. MMS provided Space Station with a messaging service that would facilitate evaluation. Since a limited set of MMS services would impact station operation, messages which invoke those services would be the only ones that need to be evaluated before they affected the station operation. The MMS message structure would have facilitated detection and analysis of the real-time flow of messages to-and-from the Space Station to safeguard its operation. A Space Messaging Service based on MMS will provide the same utility.

The Evolution of the Space Messaging Service

MMS was developed for the factory floor with communications over local area networks (LANs) and engineers available for emergency intervention. A Space Messaging Service would begin with MMS and evolve services that address the unique requirements for operating in space. A few of the issues that must be addressed are:

- All MMS messages are encoded using the ISO Basic Encoding Rules (BER). The selection of BER for the manufacturing environment provides a robust solution for heterogeneous systems that are constantly changing and are not severely bandwidth restricted. However, the space environment remains bandwidth constrained. Since the BER are not very efficient, alternative encoding procedures such as PACK encoding will be investigated for the Space Messaging Service.

- Most MMS services are confirmed services. Confirmed services are initiated by a client and require a response from the corresponding server. While the factory environment can support these services under all situations, the space environment requires some special considerations. For example, a client application in a factory will establish a connection to a server before conducting an MMS dialog with the server. The MMS Initiate Connection service is a confirmed service and some commercial implementations of MMS require a positive response from the server before other MMS services can be invoked. Translated to the ground-to-space scenario, the ground station would initiate a connection and wait for a response from the spacecraft before reading or writing data to the spacecraft. This clearly will not serve spacecraft in deep space where a round-trip light time may be hours. A Space Messaging Service will provide services with optional confirmations to address these space communication issues.

- The File Management services specified by MMS were intended to meet the limited requirements of the manufacturing systems on the factory floor. While the MMS file services provide most of the functions required for the space environment, additional services will be added to the Space Messaging Service to support a full network file server.

- An emergency on the factory floor can be addressed by a plant engineer. Short of the Space Station with an onboard astronaut, an emergency aboard a spacecraft must be addressed remotely. Therefore, the Space Messaging Service must provide services to support spacecraft emergency situations.

- The Consultative Committee for Space Data Systems (CCSDS) Packet Telemetry standards have become the standard for space data systems around the world. The Space Messaging Service will provide services designed specifically to support the CCSDS Telemetry standards.

The Space Messaging Service will be designed in a modular fashion that will permit the implementor of a spacecraft system to select only those services required to support a specific device. For example, an intelligent MMS device can be built using only a handful of the 87 MMS services. Only the software required to support the limited MMS services is incorporated into the device making it a small and efficient software implementation.

Summary

The days of hand-crafted spacecraft are coming to a close. The current budget environment will support only a limited number of custom, one-of-a-kind spacecraft. Drastic action must be taken to reduce the cost of building and flying spacecraft if space research and exploration is to continue. While space may be new, the problem of reducing the cost of things is not. In the early days of this century, only the wealthy could afford automobiles. Every automobile was a hand-crafted work of art. When Henry Ford decided to build a car that everyone could afford, the assembly line was invented. But assembly line production required interchangeable parts, and interchangeable parts required standard measurements. Consequently, the Ford standards laboratory became a critical part of the automobile plant. While the target and problems may be different, the general solution remains standard interfaces.

A Space Messaging Service would provide a standard data system interface for spacecraft and spacecraft devices. Manufacturing Message Specification provides an excellent starting point for the development of such a standard. Designed to support the automation of robots and intelligent devices in factories, MMS provides 87 communication services to build automated, distributed systems. The successful application of MMS factories around the world provides confidence that a sophisticated messaging service can be constructed and used for automation of robotic systems. Manufacturers have used MMS to build equipment that can interface and inter-operate with equipment and software from other manufacturers, without the exchange of software elements such as intermediate definition languages that must be re-compiled and re-linked. Installation of MMS devices involves the creation or configuration of tables or files. Manufacturers safeguard their proprietary interests by delivering executables instead of source code. And because installation does not require the integration, compilation and linking of software, manufacturers can maintain tight control over the behavior of their equipment or instruments.

A Space Messaging Service would provide these same benefits for the construction and operation of spacecraft. The manufacturers of spacecraft and spacecraft components will be able to focus on building devices instead of unique interfaces. A standard Space Messaging Service will promote the development of software for use on mulitple spacecraft. In addition, a standard will provide ground support facilities with a common basis for operating spacecraft. And, the reliability of spacecraft systems and support facilities will be enhanced as components are used and reused over multiple missions.

References

1. Electronic Industries Association - 511, "Manufacturing Message Specification", 1989.

2. International Organization for Standards, ISO 9506, Information Processing Systems - Open Systems Interconnection - Manufacturing Message Specification, 1989.

3. Heuser, Wm. Randy, "An OSI Architecture for the Deep Space Network", TDA Progress Report 42-109, Jet Propulsion Laboratory, Pasadena, CA, March 30, 1992.

4. Heuser, Wm. Randy, and Cooper, Lynne P., "An OSI Architecture for the DSN," SpaceOps '92, Pasadena, CA, November 1992.

5. Tang, A. & Scoggins, S; Open Networking with OSI, Prentice-Hall, Incorporated, 1992.

6. Heuser, Wm. Randy, Chen, Richard, and Stockett, Michael H.; Using Manufacturing Message Specification for Monitor and Control at Venus, Network Technology Conference, NASA Conference Publication 3241, 1993.

7. Schulz, Klaus-Jurgen; Service Management of a Spacecraft Ground Station Network in Support of Spacecraft Operations, Integrated Network Management, III (C-12), 1993 IFIP.

Appendix A. List of MMS Services

1. Virtual Manufacturing Device (VMD) provide the basic services to management a remote device.

   Initiate
   Conclude
   Cancel
   Abort
   Status
   Unsolicited Status
   Get Name List
   Identify
   Rename
   Get Capability List

2. Domain management provides services to manage a named group of resources within a VMD.

   InitiateDownSequence
   DownloadSegment
   TerminateDownSequence
   InitiateUploadSequence
   UploadSegment
   TerminateUploadSequence
   RequestDomainDownload
   RequestDomainUpload
   LoadDomainContent
   StoreDomainContent
   DeleteDomain
   GetDomainAttributes

3. Program Invocation management provides services to control programs running on a VMD.

   CreateProgramInvocation
   DeleteProgramInvocation
   Start
   Stop
   Resume
   Reset
   Kill
   GetProgramInvocationAttributes

4. Variable Access provides the services to manage variables on VMD.

   Read
   Write
   InformationReport
   GetVariableAccessAttributes
   DefineNamedVariable
   DefineScatteredAccess
   GetScatteredAccessAttributes
   DeleteVariableAccess
   DefineNamedVariableList
   GetNamedVariableListAttributes
   DeleteNamedVariableList
   DefineNamedType
   GetNamedTypeAttributes
   DeleteNamedType

5. Semaphore management provides services which allow synchronization, control and coordination of shared resources between VMDs.

   TakeControl
   RelinquishControl
   DefineSemaphore
   DeleteSemaphore
   ReportSemaphoreStatus
   ReportPoolSemaphoreStatus
   ReportSemaphoreEntryStatus
   AttachToSemaphoreModifier

6. Operator Communications provide services for the exchange of text messages between terminals on the factory floor.

   Input
   Output

7. Event management services provide facilities which allow a client MMS-user to define and manage event objects at a VMD and obtain notification of event occurrences.

   DefineEventCondition
   DeleteEventCondition
   GetEventConditionAttributes
   ReportEventConditionStatus
   AlterEventConditionMonitoring
   TriggerEvent
   DefineEventAction
   DeleteEventAction
   GetEventActionAttributes
   ReportEventActionStatus
   DefineEventEnrollment
   DeleteEventEnrollment
   GetEventEnrollmentAttributes
   ReportEventEnrollmentStatus
   AlterEventEnrollment
   EventNotification
   AcknowledgeEventEnrollment
   GetAlarmSummary
   GetAlarmEnrollmentSummary
   AttachToEventConditionModifier

8. Journal management provides services for recording and retrieval of chronologically ordered information concerning events, variable contents of interest in conjunction with events, and textual strings entered by an operator as information.

   ReadJournal
   WriteJournal
   InitializeJournal
   ReportJournalStatus
   CreateJournal
   DeleteJournal

9. File management provides FTAM services for MMS.

   ObtainFile
   FileOpen
   FileRead
   FileClose
   FileRename
   FileDelete
   FileDirectory

Appendix A. List of MMS Services

1. Virtual Manufacturing Device (VMD) provide the basic services to management a remote device.

   Initiate
   Conclude
   Cancel
   Abort
   Status
   Unsolicited Status
   Get Name List
   Identify
   Rename
   Get Capability List

2. Domain management provides services to manage a named group of resources within a VMD.

   InitiateDownSequence
   DownloadSegment
   TerminateDownSequence
   InitiateUploadSequence
   UploadSegment
   TerminateUploadSequence
   RequestDomainDownload
   RequestDomainUpload
   LoadDomainContent
   StoreDomainContent
   DeleteDomain
   GetDomainAttributes

3. Program Invocation management provides services to control programs running on a VMD.

   CreateProgramInvocation
   DeleteProgramInvocation
   Start
   Stop
   Resume
   Reset
   Kill
   GetProgramInvocationAttributes

4. Variable Access provides the services to manage variables on VMD.

   Read
   Write
   InformationReport
   GetVariableAccessAttributes
   DefineNamedVariable
   DefineScatteredAccess
   GetScatteredAccessAttributes

DeleteVariableAccess
DefineNamedVariableList
GetNamedVariableListAttributes
DeleteNamedVariableList
DefineNamedType
GetNamedTypeAttributes
DeleteNamedType

5. Semaphore management provides services which allow synchronization, control and coordination of shared resources between VMDs.

    TakeControl
    RelinquishControl
    DefineSemaphore
    DeleteSemaphore
    ReportSemaphoreStatus
    ReportPoolSemaphoreStatus
    ReportSemaphoreEntryStatus
    AttachToSemaphoreModifier

6. Operator Communications provide services for the exchange of text messages between terminals on the factory floor.

    Input
    Output

7. Event management services provide facilities which allow a client MMS-user to define and manage event objects at a VMD and obtain notification of event occurrences.

    DefineEventCondition
    DeleteEventCondition
    GetEventConditionAttributes
    ReportEventConditionStatus
    AlterEventConditionMonitoring
    TriggerEvent
    DefineEventAction
    DeleteEventAction
    GetEventActionAttributes
    ReportEventActionStatus
    DefineEventEnrollment
    DeleteEventEnrollment
    GetEventEnrollmentAttributes
    ReportEventEnrollmentStatus
    AlterEventEnrollment
    EventNotification
    AcknowledgeEventEnrollment
    GetAlarmSummary
    GetAlarmEnrollmentSummary
    AttachToEventConditionModifier

8. Journal management provides services for recording and retrieval of chronologically ordered information concerning events, variable contents of interest in conjunction with events, and textual strings entered by an operator as information.

ReadJournal
WriteJournal
InitializeJournal
ReportJournalStatus
CreateJournal
DeleteJournal

9. File management provides FTAM services for MMS.

ObtainFile
FileOpen
FileRead
FileClose
FileRename
FileDelete
FileDirectory